RCNN-CTC Model for Handwritten Text Recognition
In-depth Analysis of the IAM Dataset

Inhouse Project
Review - III

Team Members :

Anurag Shirsekar (D12A-59)
Sai Thikekar (D12A-64)
Uzair Shaikh (D12A-56)
Yash Chhaproo (D12A-07)

Project Mentor : **Mrs. Lifna C. S.**

V.E.S.

Group no: 29

Group Number :
Group Members :

Content

I. Introduction to Project

II. Lacuna in the existing systems

III. Problem Definition

IV. Literature Survey

V. Methodology employed

VI. Hardware, Software, tools and the constraints

VII. Modular Diagram

VIII. Results and Discussion

IX. Conclusion

X. References

Introduction to Project

- Handwritten text recognition (HTR) is a vital field within pattern recognition and deep learning, dedicated to converting handwritten text into digital format. In this presentation, we conduct a thorough analysis of HTR techniques using the IAM dataset. Our primary focus is on assessing the performance of three key models: **CNN-BiGRU, CNN-BiLSTM, and RCNN-CTC**.

- By evaluating these models, we aim to tackle the challenges associated with accurate character recognition and word transcription in handwritten documents. Through our investigation, we aim to offer valuable insights into the capabilities and limitations of each model, thereby contributing to the advancement of HTR methodologies.

Lacuna in the Existing system

Name of Existing system	Lacunas	Improvement required
Tesseract OCR	Low Accuracy for Handwritten Text Recognition, Generalised OCR, CNN Based Architecture.	To improve the accuracy for Handwritten Text Recognition we can finetune the CNN architecture by exploring other approaches
EasyOCR	Generalised OCR, Architecture is a combination of RNN and CNN	Including RNN-CNN at the CTC decoder can improve the accuracy for text Recognition.
Adobe Acrobat OCR	Low Accuracy for Handwritten Text Recognition	CNN architecture is employed along with language model.

Problem Definition

Handwritten Text Recognition (HTR): The project focuses on the task of converting handwritten text into digital format.

Challenges: Handwritten text recognition poses challenges due to variations in handwriting styles, deformations, and noise in handwritten documents.

Evaluation of Models: Three distinct HTR models—CNN-BiGRU, CNN-BiLSTM, and RCNN-CTC—are evaluated using the IAM dataset.

Objectives:

Assess Accuracy: Evaluate the accuracy of character recognition and word transcription.

Problem Definition

Identify Strengths and Limitations: Identify the strengths and limitations of each HTR model.

Contribution to Advancements: Contribute insights to advance methodologies in HTR.

Contribution: The project aims to enhance the efficiency and accuracy of handwritten text recognition systems by addressing these challenges and providing valuable insights into the performance of different HTR models.

Literature Survey

Paper Number	**Algorithm and Features**	**Methodology**	**Evaluation Measures and Analysis**
Paper [1]	● CONVOLUTIONAL NEURAL NETWORK (CNN) Based Model	● Preprocessing Techniques ● Generalised Layers	● The paper presents a CNN based neural network architecture for handwritten text recognition. ● Utilizing convolutional layers,the model works for the particular dataset
Paper [2]	Stacey Whitmore	Operating Procedure Extender for Novel Systems (OPENS)	The concepts proven by OPENS could automate the process of converting PBPs to CBPs, significantly reducing the time and cost involved.

Literature Survey

Paper Number	Algorithm and Features	Methodology	Evaluation Measures and Analysis
Paper [3]	Baoguang Shi, Xiang Bal, Cong Yao	PIL (Python Imaging Library) or Pillow, Matplotlib, OpenCV, Pytesseract	The paper presents a powerful neural network architecture for scene text recognition.Utilizing convolutional and recurrent layers,the model outperforms existing methods, advancements in real-world text recognition.
Paper [4]	CNN and OpenCV libraries for data processing, Open source OCR engine	Data Preprocessing on images, use tesseract to extract data and then output in the form of csv	Struggles with handwritten bills. Object misidentification. Emphasizing the need for improved OCR techniques across document types.

Literature Survey

Paper Number	Algorithm and Features	Methodology	Evaluation Measures and Analysis
Paper [5]	DNN model for table prediction	A verification phase with graph representation which improved precision by reducing false positives.	Graph representation, effectively reduced false positives and significantly increased the overall precision of table detection
Paper [6]	Minghui Liao1, Zhaoyi Wan, Cong Yao, Kai Chen, Xiang Bai	Real-time Scene Text Detection with Differentiable Binarization.	Algorithm used for Text detection.

Literature Survey

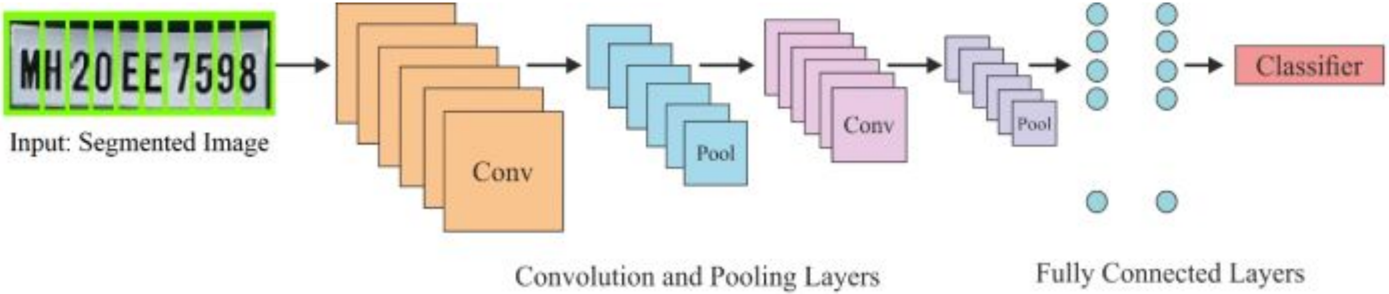
Paper Number	Algorithm and Features	Methodology	Evaluation Measures and Analysis
Paper [7]	Firat Kizilirmak , Berrin Yanikoglu	CNN-BiLSTM model for English Handwriting Recognition: Comprehensive Evaluation on the IAM Dataset	CNN-BiLSTM with CTC achieves 3.59% CER, proposes test-time augmentation, error analysis, and releases public code for reproducibility.

Methodology Employed

CLASSIFIERS

1. **Convolutional Neural Network (CNN):**

- CNN is a key algorithm in deep learning, particularly for image processing tasks.
- It operates by taking an image as input, extracting features, and distinguishing between them.
- Inspired by the connections of neurons in the human brain, CNN learns to detect features through training data.



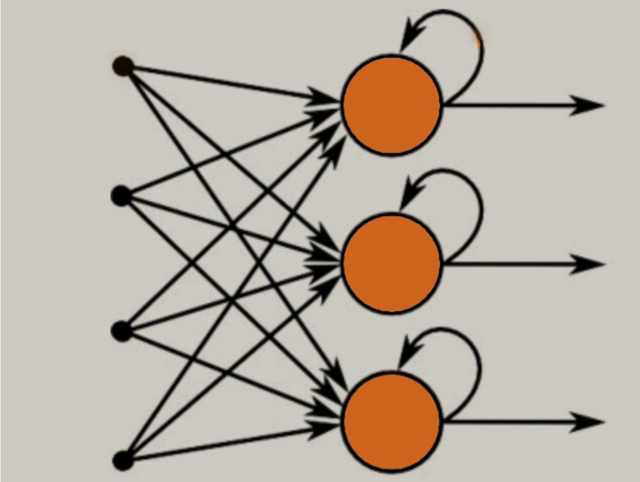
Methodology Employed

Recurrent Neural Network (RNN):

- RNN is widely used in deep learning for sequential data processing tasks like speech and text recognition.

- It performs the same task for each element in the data sequence, and the output depends on previous outputs.

- RNNs are suitable for tasks where the input and output have a temporal relationship.

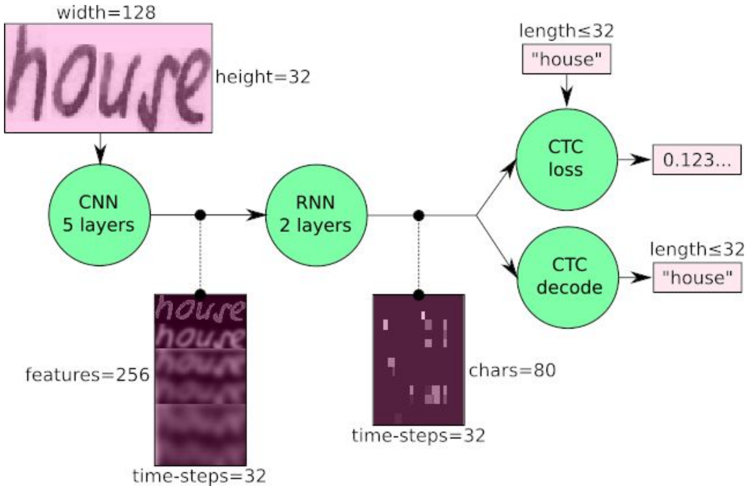


Connectionist Temporal Classification (CTC):

- CTC introduces a differentiable cost function for training RNNs to identify and label unsegmented sequences.

- It includes an additional blank symbol in the possible labels, enabling RNNs to output probabilities over all labels.

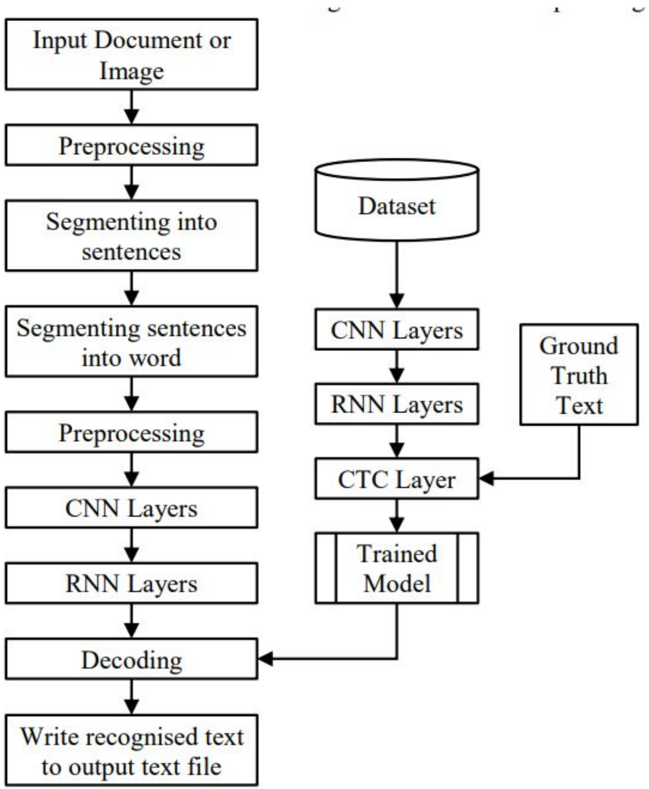
- CTC is particularly useful for sequence labeling tasks such as speech recognition and handwriting recognition.



PROPOSED CRNN-CTC MODEL

1. Input Image Processing:

- Input image fed into CNN layers for featur extraction.
- CNN performs convolution, activation (ReLU and downsizing operations.
- 5x5 and 3x3 filter kernels used for different layers
- Pooling layer downsizes image height by 2 and adds channels for feature mapping.



2. Feature Sequence Generation:

- Output sequence of 32x256 features generated.
- Each feature sequence timestep applied to an LSTM network.

3. Long Short-Term Memory (LSTM) Network:

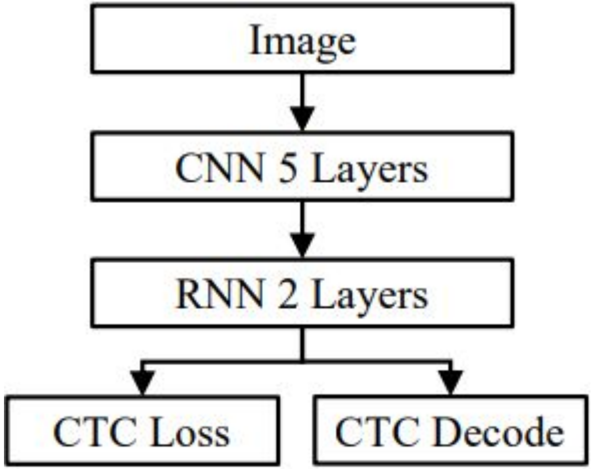
- **LSTM chosen for superior data transfer and training capabilities over Vanilla RNN.**
- **LSTM output sequence mapped to a 32x80 matrix.**
- **79 characters from IAM dataset and an additional character for CTC blank labels.**

4. Connectionist Temporal Classification (CTC):

- **Ground truth text and RNN output matrix fed into CTC layer during training.**
- **CTC decodes output matrix into text, compares with ground truth, and computes loss.**
- **RMSProp optimizer used for training based on average loss values.**

5. Model Training and Evaluation:

- **Trained model achieves a word error rate (WER) of 10.62%.**
- **Trained model utilized for recognizing input images.**



Image

CNN 5 Layers

RNN 2 Layers

CTC Loss

CTC Decode

Hardware, Software, Tools and constraint

Hardware

- Google Colab V100 GPU
- 32GB RAM and 8GB Nvdia GPU
- Inteli7 or Ryzen 5 processor

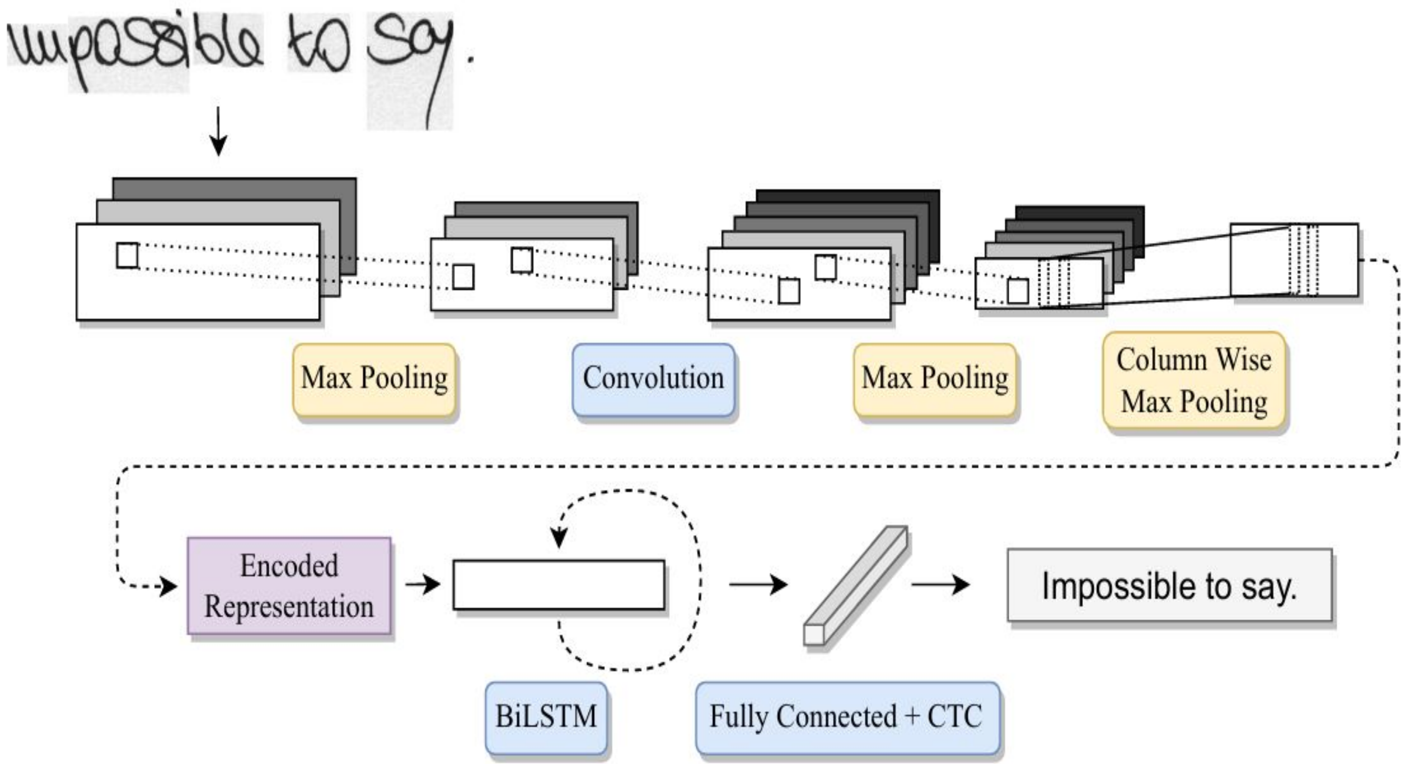
Software

- IDE like VS code/ PyCharm
- Jupyter notebook
- Tensorflow
- Keras

Python libraries:

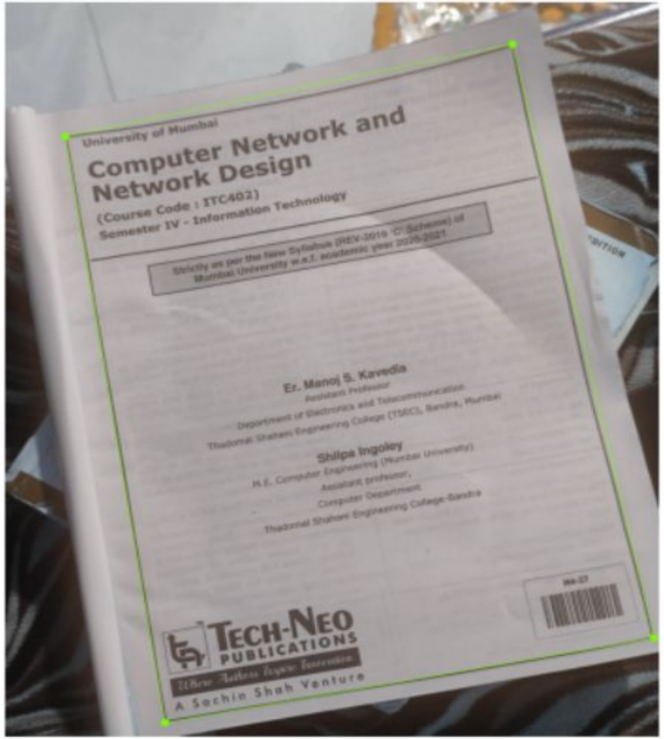
- Cv2
- matplotlib

Modular Diagram



IMPLEMENTATION

Input Image:



University of Mumbai

Computer Network and Network Design

(Course Code : ITC402)

Semester IV - Information Technology

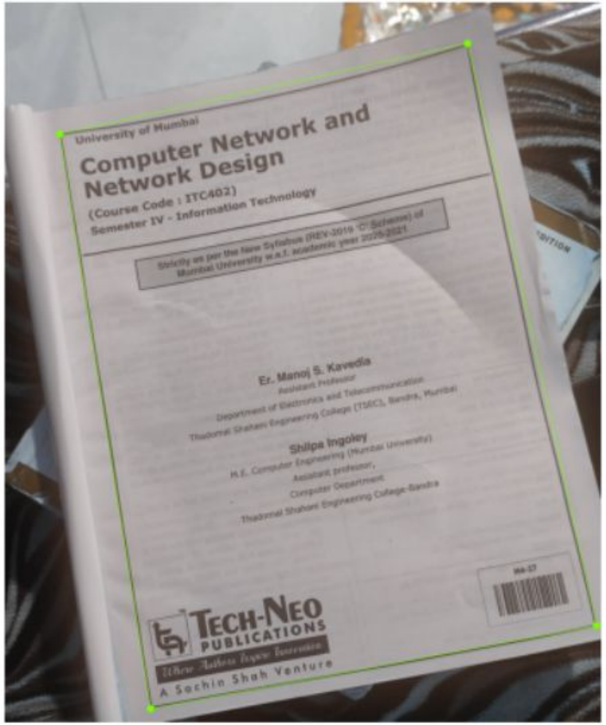
Strictly as per the New Syllabus (REV-2019 'C' Scheme) of Mumbai University w.e.f. academic year 2020-2021

Er. Manoj S. Kavedia
Assistant Professor
Department of Electronics and Telecommunication
Thadomal Shahani Engineering College (TSEC), Bandra, Mumbai

Shilpa Ingoley
M.E. Computer Engineering (Mumbai University)
Assistant professor,
Computer Department
Thadomal Shahani Engineering College-Bandra

TECH-NEO PUBLICATIONS
Where Authors Inspire Innovation
A Sachin Shah Venture

Contour Detection:



University of Mumbai

Computer Network and Network Design

(Course Code : ITC402)

Semester IV - Information Technology

Strictly as per the New Syllabus (REV-2019 'C' Scheme) of Mumbai University w.e.f. academic year 2020-2021

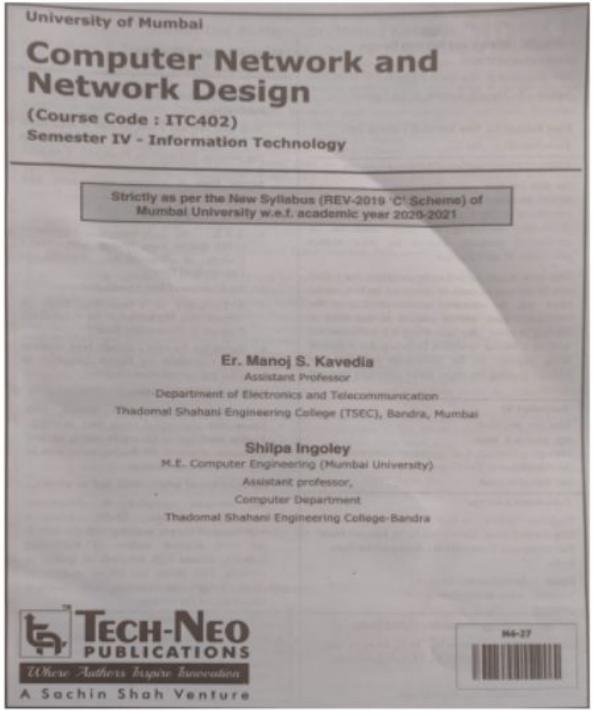
Er. Manoj S. Kavedia
Assistant Professor
Department of Electronics and Telecommunication
Thadomal Shahani Engineering College (TSEC), Bandra, Mumbai

Shilpa Ingoley
M.E. Computer Engineering (Mumbai University)
Assistant professor,
Computer Department
Thadomal Shahani Engineering College-Bandra

TECH-NEO PUBLICATIONS
Where Authors Inspire Innovation
A Sachin Shah Venture

IMPLEMENTATION

Preprocessing of image:



University of Mumbai

Computer Network and Network Design

(Course Code : ITC402)

Semester IV - Information Technology

Strictly as per the New Syllabus (REV-2019 'C' Scheme) of Mumbai University w.e.f. academic year 2020-2021

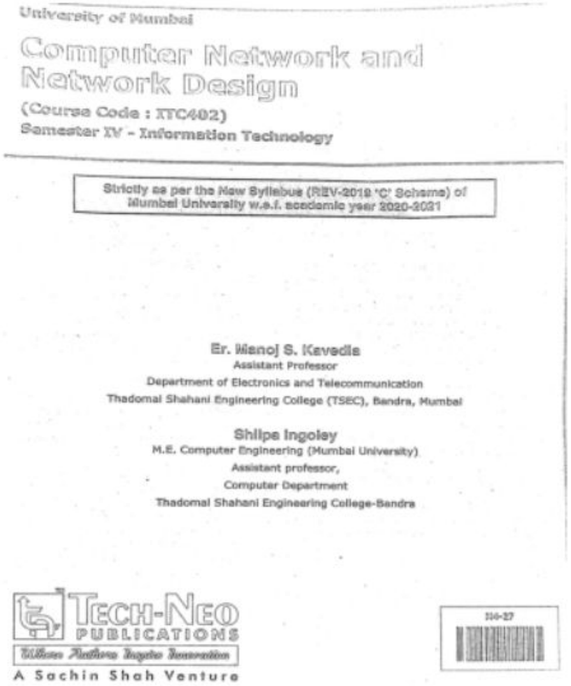
Er. Manoj S. Kavedia
Assistant Professor
Department of Electronics and Telecommunication
Thadomal Shahani Engineering College (TSEC), Bandra, Mumbai

Shilpa Ingoley
M.E. Computer Engineering (Mumbai University)
Assistant professor,
Computer Department
Thadomal Shahani Engineering College-Bandra

TECH-NEO PUBLICATIONS
Where Authors Inspire Innovation
A Sachin Shah Venture

M4-27

Enhancing the quality of image:



University of Mumbai

Computer Network and Network Design

(Course Code : ITC402)

Semester IV - Information Technology

Strictly as per the New Syllabus (REV-2019 'C' Scheme) of Mumbai University w.e.f. academic year 2020-2021

Er. Manoj S. Kavedia
Assistant Professor
Department of Electronics and Telecommunication
Thadomal Shahani Engineering College (TSEC), Bandra, Mumbai

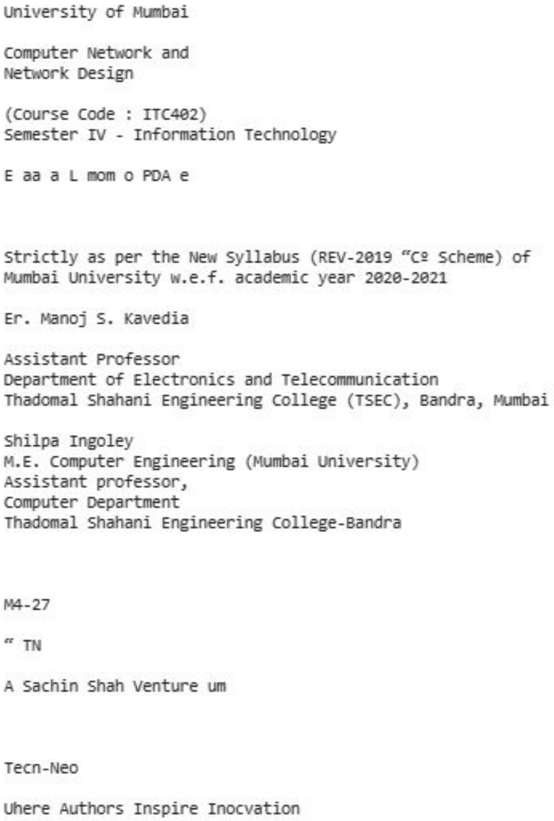
Shilpa Ingoley
M.E. Computer Engineering (Mumbai University)
Assistant professor,
Computer Department
Thadomal Shahani Engineering College-Bandra

TECH-NEO PUBLICATIONS
Where Authors Inspire Innovation
A Sachin Shah Venture

M4-27

IMPLEMENTATION

Output:

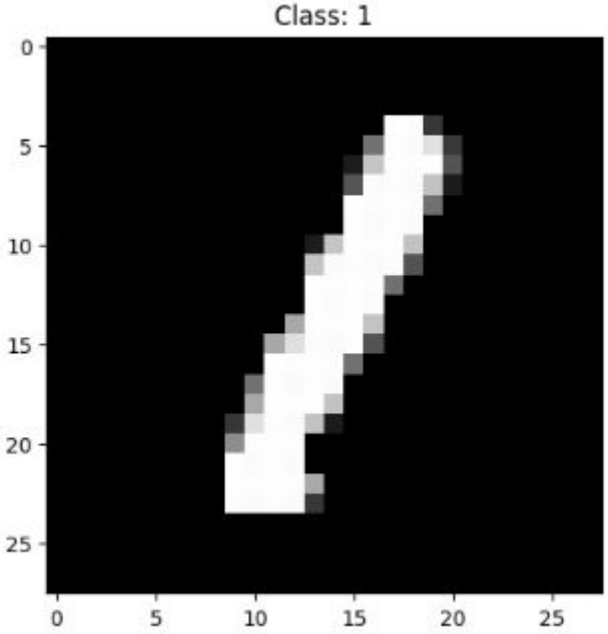


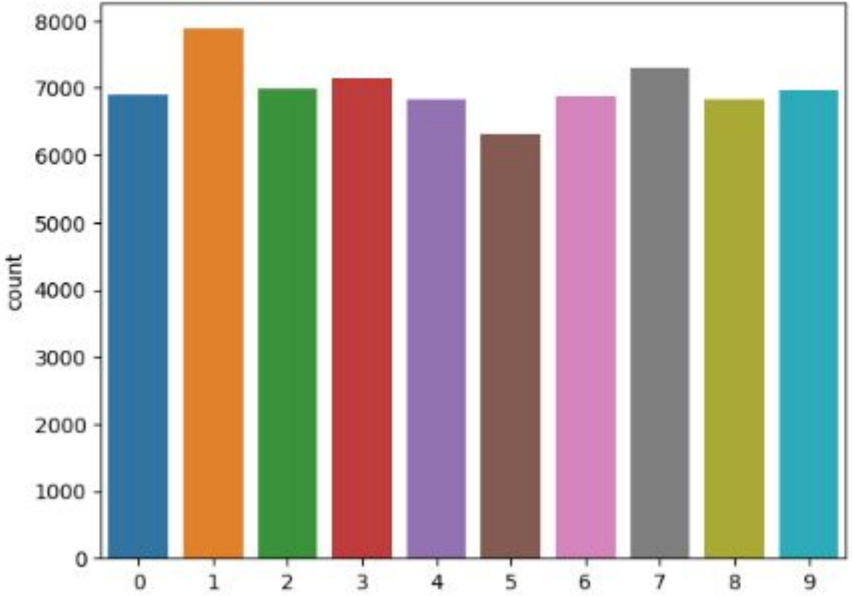
```
University of Mumbai

Computer Network and
Network Design

(Course Code : ITC402)
Semester IV - Information Technology

E aa a L mom o PDA e



Strictly as per the New Syllabus (REV-2019 “Cº Scheme) of
Mumbai University w.e.f. academic year 2020-2021

Er. Manoj S. Kavedia

Assistant Professor
Department of Electronics and Telecommunication
Thadomal Shahani Engineering College (TSEC), Bandra, Mumbai

Shilpa Ingoley
M.E. Computer Engineering (Mumbai University)
Assistant professor,
Computer Department
Thadomal Shahani Engineering College-Bandra



M4-27

“ TN

A Sachin Shah Venture um



Tecn-Neo

Uhere Authors Inspire Inocvation
```

IMPLEMENTATION

Training of Custom OCR

MNIST 0-9 dataset

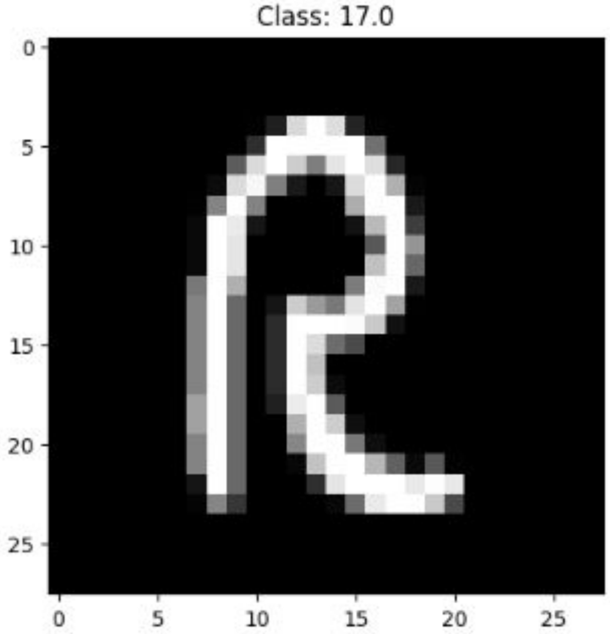


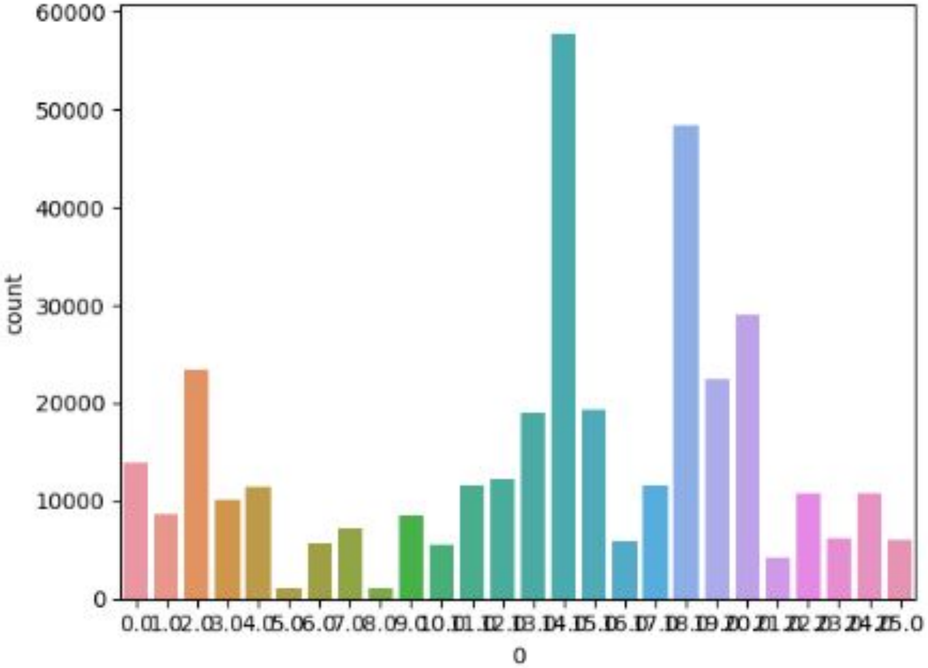


IMPLEMENTATION

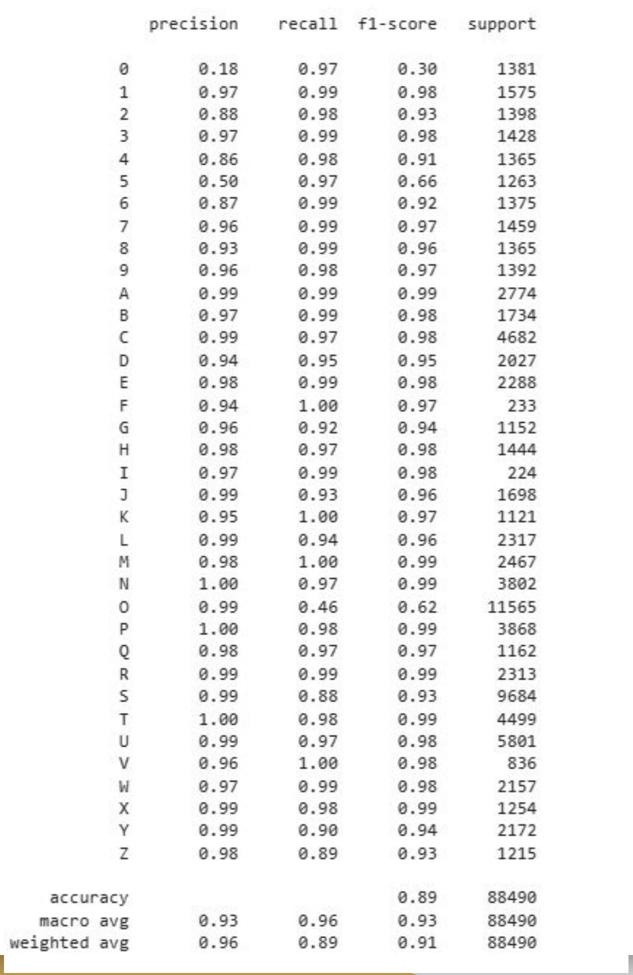
Training of Custom OCR

Kaggle A-Z dataset





IMPLEMENTATION

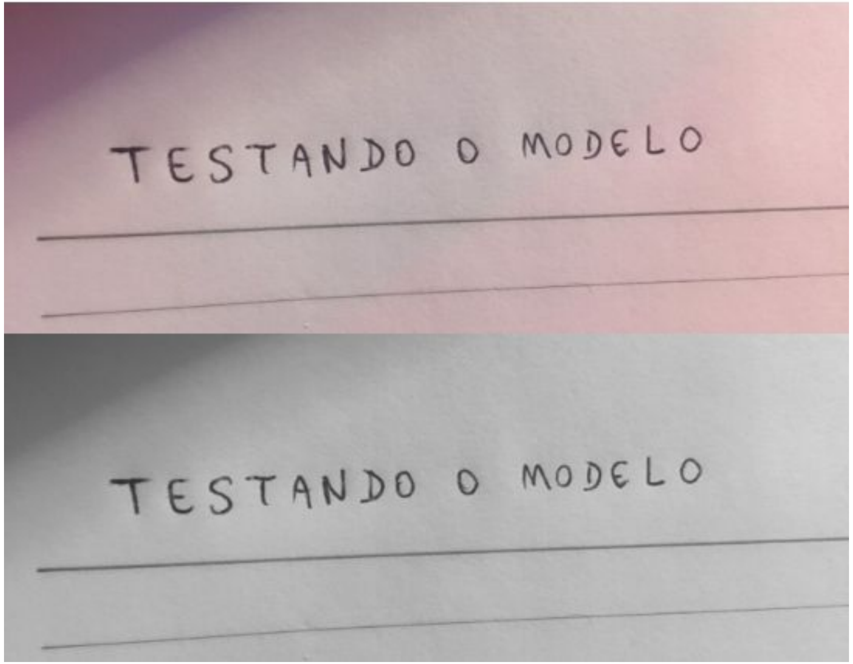


```
              precision    recall  f1-score   support

           0       0.18      0.97      0.30      1381
           1       0.97      0.99      0.98      1575
           2       0.88      0.98      0.93      1398
           3       0.97      0.99      0.98      1428
           4       0.86      0.98      0.91      1365
           5       0.50      0.97      0.66      1263
           6       0.87      0.99      0.92      1375
           7       0.96      0.99      0.97      1459
           8       0.93      0.99      0.96      1365
           9       0.96      0.98      0.97      1392
           A       0.99      0.99      0.99      2774
           B       0.97      0.99      0.98      1734
           C       0.99      0.97      0.98      4682
           D       0.94      0.95      0.95      2027
           E       0.98      0.99      0.98      2288
           F       0.94      1.00      0.97       233
           G       0.96      0.92      0.94      1152
           H       0.98      0.97      0.98      1444
           I       0.97      0.99      0.98       224
           J       0.99      0.93      0.96      1698
           K       0.95      1.00      0.97      1121
           L       0.99      0.94      0.96      2317
           M       0.98      1.00      0.99      2467
           N       1.00      0.97      0.99      3802
           O       0.99      0.46      0.62     11565
           P       1.00      0.98      0.99      3868
           Q       0.98      0.97      0.97      1162
           R       0.99      0.99      0.99      2313
           S       0.99      0.88      0.93      9684
           T       1.00      0.98      0.99      4499
           U       0.99      0.97      0.98      5801
           V       0.96      1.00      0.98       836
           W       0.97      0.99      0.98      2157
           X       0.99      0.98      0.99      1254
           Y       0.99      0.90      0.94      2172
           Z       0.98      0.89      0.93      1215

    accuracy                           0.89     88490
   macro avg       0.93      0.96      0.93     88490
weighted avg       0.96      0.89      0.91     88490
```

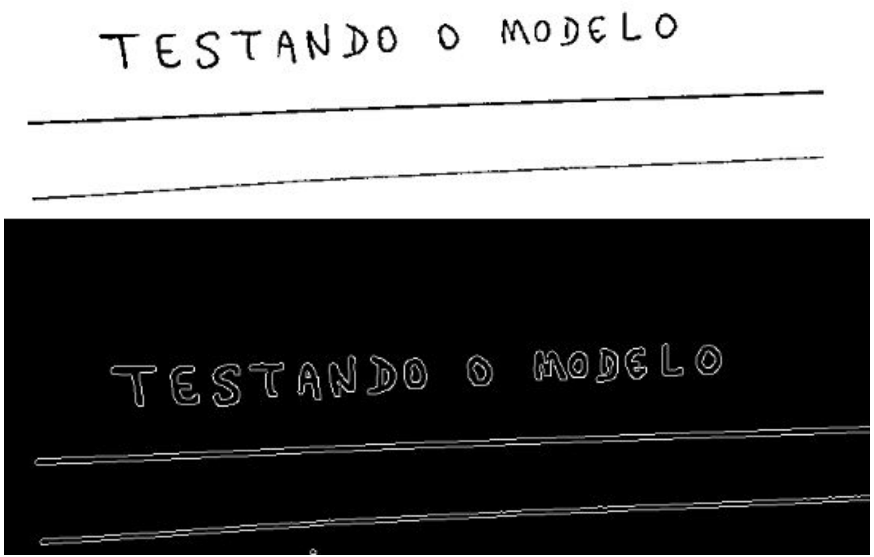
IMPLEMENTATION

Text Extraction from an image after training the neural network

Input Image



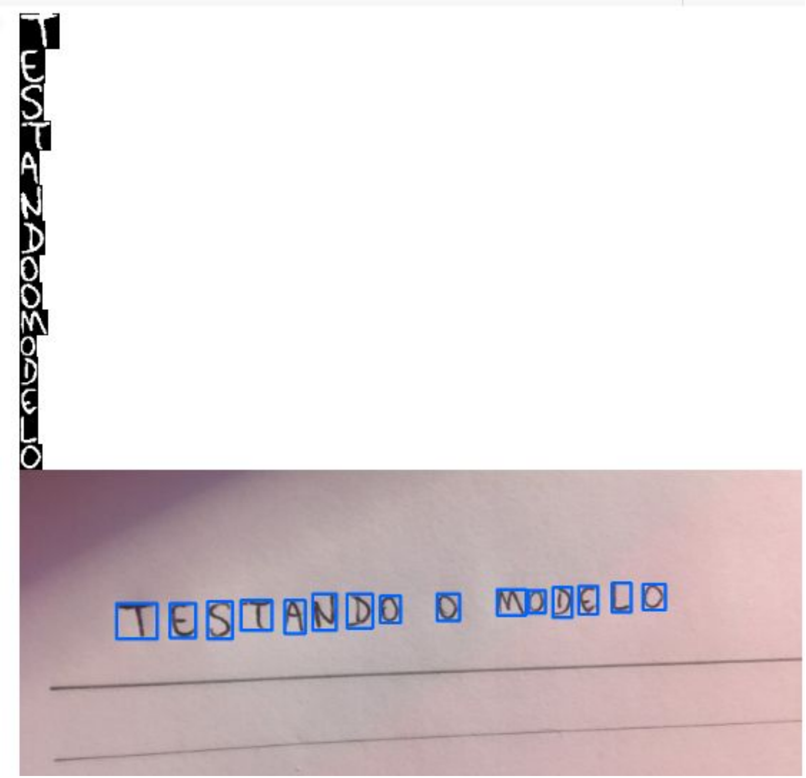
Preprocessing the image



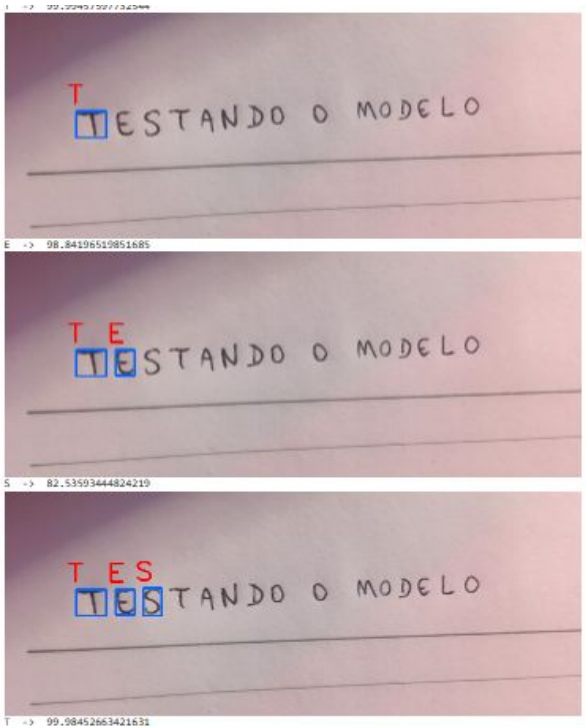
IMPLEMENTATION

Text Extraction from an image after training the neural network

Contour Detection



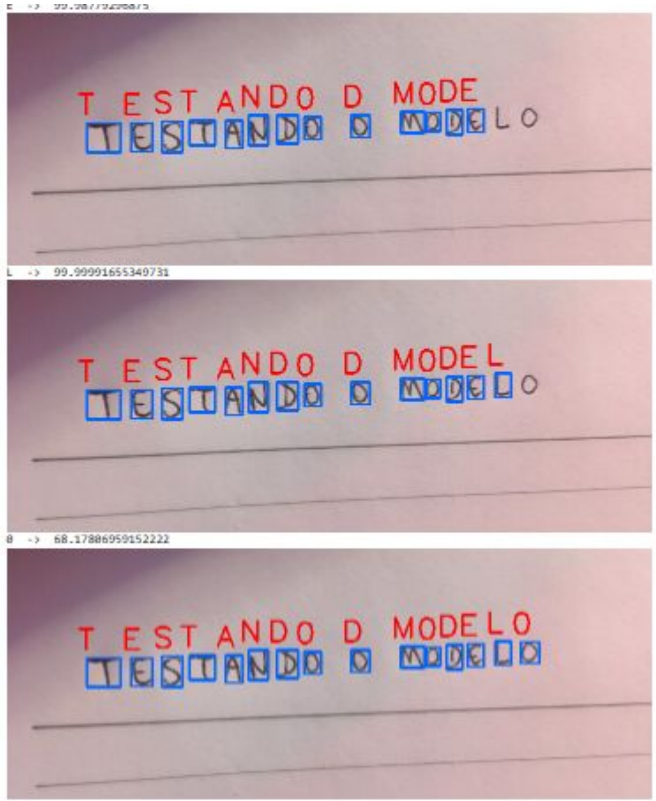
Recognition of text



```
T  ->  99.99457597732544
E  ->  98.84196519851685
S  ->  82.53593444824219
T  ->  99.98452663421631
```

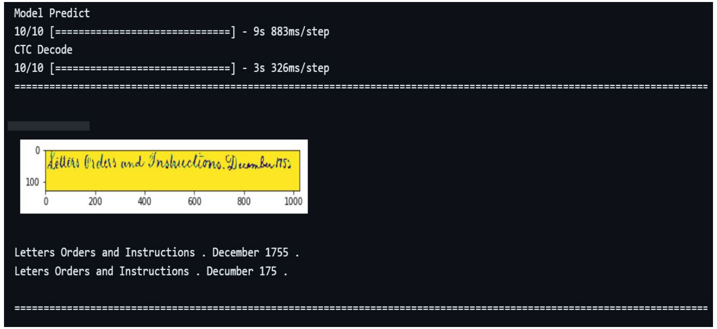
IMPLEMENTATION

Text Extraction from an image after training the neural network

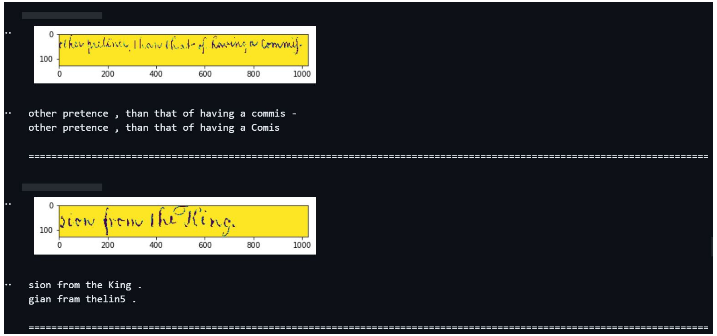


```
E  ->  99.98779296875
L  ->  99.99991655349731
0  ->  68.17886959152222
```

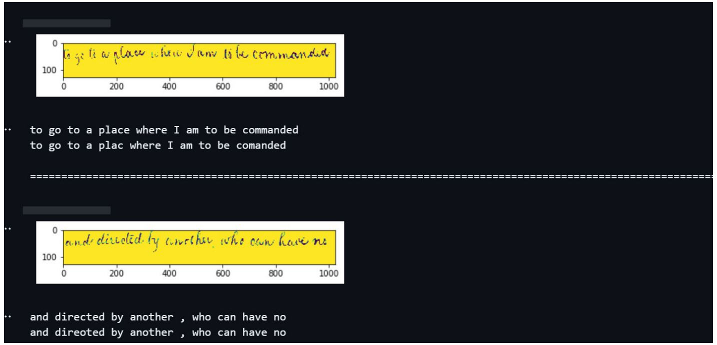
Predict



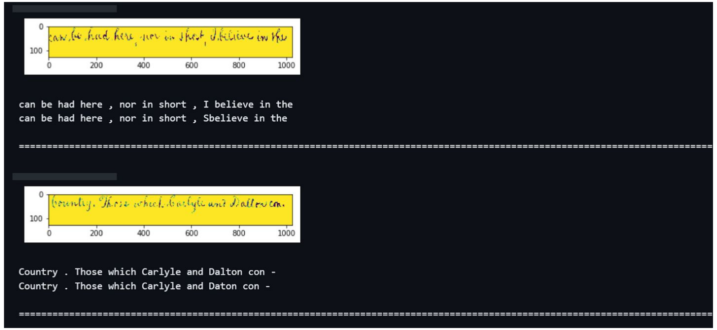
```
Model Predict
10/10 [==============================] - 9s 883ms/step
CTC Decode
10/10 [==============================] - 3s 326ms/step
================================================================================================================

Letters Orders and Instructions . December 1755 .
Leters Orders and Instructions . Decumber 175 .

================================================================================================================
```



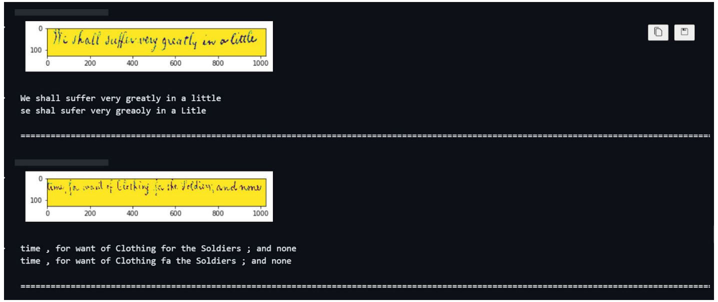
```
other pretence , than that of having a commis -
other pretence , than that of having a Comis

================================================================================================================

sion from the King .
gian fram thelin5 .

================================================================================================================
```



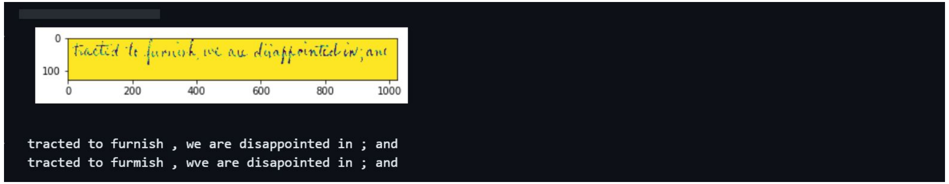
```
to go to a place where I am to be commanded
to go to a plac where I am to be comanded

================================================================================================================

and directed by another , who can have no
and direoted by another , who can have no
```



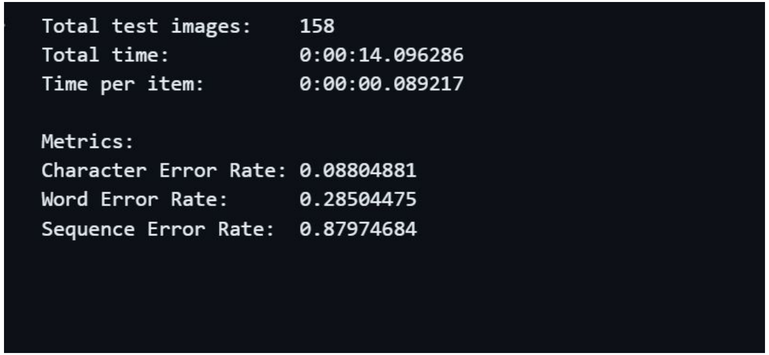
```
can be had here , nor in short , I believe in the
can be had here , nor in short , Sbelieve in the

================================================================================================================

Country . Those which Carlyle and Dalton con -
Country . Those which Carlyle and Daton con -

================================================================================================================
```



```
We shall suffer very greatly in a little
se shal sufer very greaoly in a Litle

==========================================================================================================================

time , for want of Clothing for the Soldiers ; and none
time , for want of Clothing fa the Soldiers ; and none

==========================================================================================================================
```



```
tracted to furnish , we are disappointed in ; and
tracted to furmish , wve are disapointed in ; and
```



```
Total test images:    158
Total time:           0:00:14.096286
Time per item:        0:00:00.089217

Metrics:
Character Error Rate: 0.08804881
Word Error Rate:      0.28504475
Sequence Error Rate:  0.87974684
```

CNN - BiLSTM

1. Convolutional Neural Networks (CNNs):

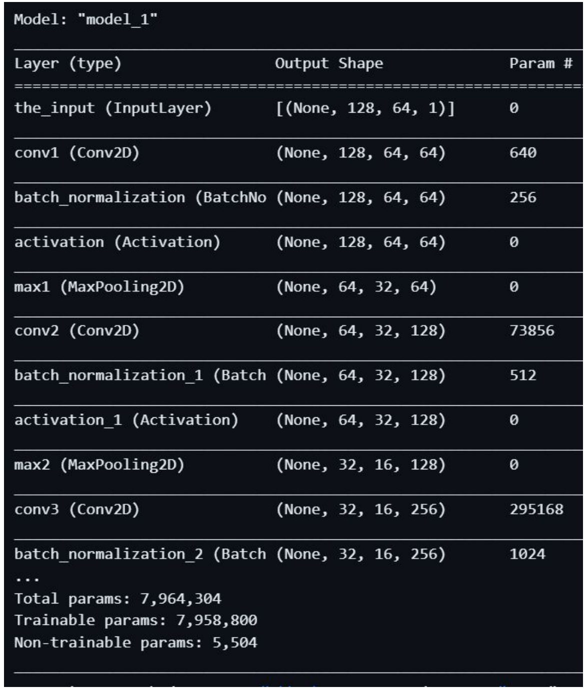
- Used for extracting spatial features from input images of handwritten text.
- CNN layers capture hierarchical patterns and structures present in the input images.

2. Bidirectional Long Short-Term Memory (BiLSTM):

- BiLSTM is a type of recurrent neural network (RNN) capable of capturing sequential dependencies in data.
- Utilized to process the feature sequences extracted by CNNs, capturing temporal information inherent in handwritten text.
- Bidirectional aspect enables the model to consider context from both past and future inputs simultaneously, enhancing recognition accuracy.

3. Application:

- CNN-BiLSTM architectures have been successfully applied in handwritten text recognition tasks, achieving state-of-the-art performance on benchmark datasets like IAM.



```
Model: "model_1"
_________________________________________________________________
Layer (type)                Output Shape              Param #
=================================================================
the_input (InputLayer)      [(None, 128, 64, 1)]      0
_________________________________________________________________
conv1 (Conv2D)              (None, 128, 64, 64)       640
_________________________________________________________________
batch_normalization (BatchNo (None, 128, 64, 64)      256
_________________________________________________________________
activation (Activation)     (None, 128, 64, 64)       0
_________________________________________________________________
max1 (MaxPooling2D)         (None, 64, 32, 64)        0
_________________________________________________________________
conv2 (Conv2D)              (None, 64, 32, 128)       73856
_________________________________________________________________
batch_normalization_1 (Batch (None, 64, 32, 128)      512
_________________________________________________________________
activation_1 (Activation)   (None, 64, 32, 128)       0
_________________________________________________________________
max2 (MaxPooling2D)         (None, 32, 16, 128)       0
_________________________________________________________________
conv3 (Conv2D)              (None, 32, 16, 256)       295168
_________________________________________________________________
batch_normalization_2 (Batch (None, 32, 16, 256)      1024
...
Total params: 7,964,304
Trainable params: 7,958,800
Non-trainable params: 5,504
_________________________________________________________________
```

CNN - BiGRU

1. **Convolutional Neural Networks (CNNs):**

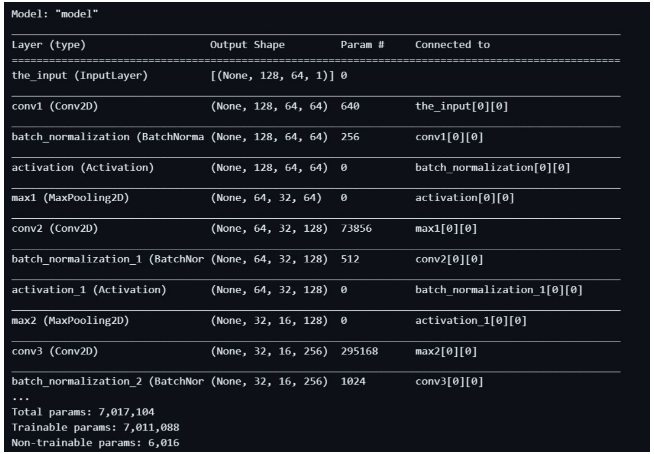
- CNNs are employed to extract spatial features from input images of handwritten text.
- These features capture hierarchical patterns and structures present in the input images.

2. **Bidirectional Gated Recurrent Units (BiGRUs):**

- BiGRUs, a variant of recurrent neural networks (RNNs), capture sequential dependencies in data.
- They process the feature sequences extracted by CNNs, capturing temporal information in handwritten text.
- Bidirectional aspect enables simultaneous consideration of context from both past and future inputs, enhancing recognition accuracy.

3. **Application:**

- CNN-BiGRU architectures have demonstrated success in handwritten text recognition tasks, achieving competitive performance on benchmark datasets such as IAM.



```
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
the_input (InputLayer)          [(None, 128, 64, 1)] 0
__________________________________________________________________________________________________
conv1 (Conv2D)                  (None, 128, 64, 64)  640         the_input[0][0]
__________________________________________________________________________________________________
batch_normalization (BatchNorma (None, 128, 64, 64)  256         conv1[0][0]
__________________________________________________________________________________________________
activation (Activation)         (None, 128, 64, 64)  0           batch_normalization[0][0]
__________________________________________________________________________________________________
max1 (MaxPooling2D)             (None, 64, 32, 64)   0           activation[0][0]
__________________________________________________________________________________________________
conv2 (Conv2D)                  (None, 64, 32, 128)  73856       max1[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 64, 32, 128)  512         conv2[0][0]
__________________________________________________________________________________________________
activation_1 (Activation)       (None, 64, 32, 128)  0           batch_normalization_1[0][0]
__________________________________________________________________________________________________
max2 (MaxPooling2D)             (None, 32, 16, 128)  0           activation_1[0][0]
__________________________________________________________________________________________________
conv3 (Conv2D)                  (None, 32, 16, 256)  295168      max2[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 32, 16, 256)  1024        conv3[0][0]
...
Total params: 7,017,104
Trainable params: 7,011,088
Non-trainable params: 6,016
```

CNN-RNN-CTC for Handwritten Text Recognition:

4. Advantages:

- End-to-end Training: The CNN-RNN-CTC architecture allows for end-to-end training of the entire model, optimizing it for handwritten text recognition without requiring intermediate steps.
- Capturing Spatial and Temporal Information: By combining CNNs for spatial feature extraction and RNNs for capturing temporal dependencies, the model effectively captures both spatial and sequential aspects of handwritten text.
- Robustness to Variability: The model's ability to capture spatial and temporal information makes it robust to variations in handwriting styles, deformations, and noise present in handwritten documents.

5. Application:

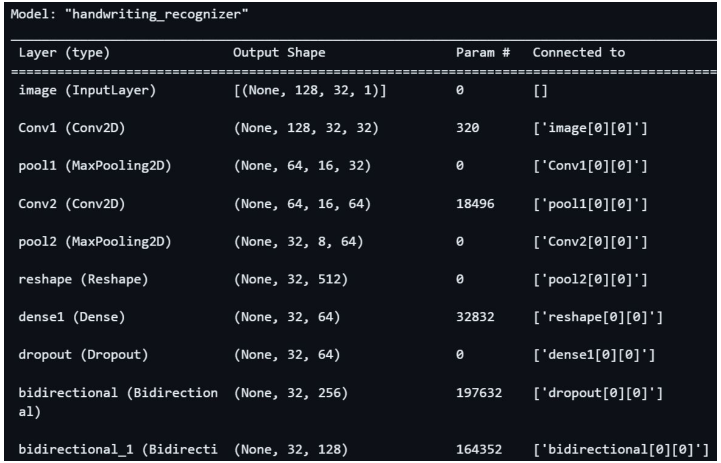
- CNN-RNN-CTC architectures have been widely used in handwritten text recognition tasks, achieving state-of-the-art performance on benchmark datasets such as IAM and RIMES.

CNN-RNN-CTC for Handwritten Text Recognition:

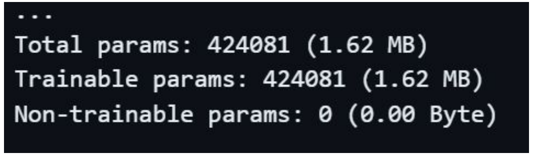
1. **Convolutional Neural Networks (CNNs):**
   - CNNs are utilized for extracting spatial features from input images of handwritten text.
   - These features capture hierarchical patterns and structures present in the input images.

2. Recurrent Neural Networks (RNNs):
   - RNNs, including variants like Long Short-Term Memory (LSTM) or Gated Recurrent Units (GRUs), capture sequential dependencies in data.
   - They process the feature sequences extracted by CNNs, capturing temporal information in handwritten text.

3. Connectionist Temporal Classification (CTC):
   - CTC is a technique used to train RNNs for sequence labeling tasks, such as speech recognition and handwritten text recognition.
   - It aligns the input sequence with the output sequence without requiring explicit alignment information.
   - Enables end-to-end training of the entire network, directly optimizing the model for sequence labeling tasks.



```
Model: "handwriting_recognizer"
__________________________________________________________________________________________________
 Layer (type)                Output Shape                 Param #   Connected to
==================================================================================================
 image (InputLayer)          [(None, 128, 32, 1)]         0         []

 Conv1 (Conv2D)              (None, 128, 32, 32)          320       ['image[0][0]']

 pool1 (MaxPooling2D)        (None, 64, 16, 32)           0         ['Conv1[0][0]']

 Conv2 (Conv2D)              (None, 64, 16, 64)           18496     ['pool1[0][0]']

 pool2 (MaxPooling2D)        (None, 32, 8, 64)            0         ['Conv2[0][0]']

 reshape (Reshape)           (None, 32, 512)              0         ['pool2[0][0]']

 dense1 (Dense)              (None, 32, 64)               32832     ['reshape[0][0]']

 dropout (Dropout)           (None, 32, 64)               0         ['dense1[0][0]']

 bidirectional (Bidirection  (None, 32, 256)              197632    ['dropout[0][0]']
 al)

 bidirectional_1 (Bidirecti  (None, 32, 128)              164352    ['bidirectional[0][0]']
```



```
...
Total params: 424081 (1.62 MB)
Trainable params: 424081 (1.62 MB)
Non-trainable params: 0 (0.00 Byte)
```

Results and Discussion

Sr. No	HTR Models	Time/Epochs	Loss	Val_Loss
1.	CNN - BiGRU	30	0.2128	0.4485
2.	CNN - BiLSTM	30	0.1281	0.2998
3.	RCNN - CTC	15	1.6617	1.7420

Conclusion

1. Model Evaluation: We assessed the performance of three models—CNN-BiGRU, CNN-BiLSTM, and CNN-RNN-CTC— for handwritten text recognition.

2. Effective Recognition: Our experiments demonstrated the effectiveness of each model in accurately recognizing handwritten text.

3. Importance of Hybrid Architectures: The combination of spatial feature extraction (CNN) and sequential processing (BiGRU, BiLSTM, RNN-CTC) proved crucial for achieving high recognition accuracy.

4. Valuable Insights: Our findings provide valuable insights into the strengths and limitations of different architectures for handwritten text recognition tasks.

5. Advancing the Field: By showcasing the capabilities of hybrid CNN-RNN architectures, we contribute to the ongoing advancement of handwritten text recognition methodologies.

6. Potential Impact: The improved accuracy and efficiency of our models hold promise for real-world applications, including automated document processing and accessibility technologies.

7. Continuing Innovation: Moving forward, further exploration and refinement of these architectures will continue to drive progress in the field of handwritten text recognition.

References

[1] Atman Mishra, A. Sharath Ram, Kavyashree C. *“Handwritten Text Recognition Using Convolutional Neural Network”*, JETIR, July, 2023

[2] A. Chawla, A. Gupta, M. Mohana, and K. S. Sushruta, "Intelligent Information Retrieval: Techniques for Character Recognition And Structured Data Extraction." -2022

[3] S. Whitmore, "Procedure Parsing: A Method for Parsing Handwritten Documents into Computer-Based Procedures," -2020

[4] B. Shi, X. Bal, and C. Yao, "An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition" -2020

[5] Alan Jiju, Shaun Tuscano and Chetana Badgujar3 Student, Department of IT, Fr. Conceicao Rodrigues Institute of Technology, Vashi, Navi Mumbai, INDIA, “OCR Text Extraction” -2021

References

[6] A. Shigarov, A. Altaev, A. Mikhailov, V. Paramonov, and E. Cherkashin, "TabbyPDF: Web-Based System for PDF Table Extraction” -2019

[7] Minghui Liao1, Zhaoyi Wan, Cong Yao, Kai Chen, Xiang Bai, “Real-time Scene Text Detection with Differentiable Binarization.”-2019

[8] CNN-BiLSTM model for English Handwriting Recognition:Comprehensive Evaluation on the IAM Dataset Firat Kizilirmak and Berrin Yanikoglu ,Faculty of Engineering and Nat. Sciences, Sabanci University, Istanbul, Turkiye, 34956.Center of Excellence in Data Analytics (VERIM), Istanbul, Turkiye, 34956.-2022